



Universidade Federal da Grande Dourados - UFGD

Faculdade de Engenharia - FAEN

Curso de Engenharia Mecânica - Bacharelado

Combustão e Combustíveis

Equilíbrio Químico

Dourados 2018

Engenheiro Responsável: Adrian Beppu Hirata

Engenheiro Verificador: Carlos Renan Cândido da Silva

Combustão e Combustíveis

Equilíbrio Químico

Dourados, 12 de novembro de 2018

Trabalho 5 – Equilíbrio Químico

A partir das instruções dada em sala, utilizou-se o **Scilab versão 5.5.2** para poder realizar o Trabalho 5 que é dividido em 2 partes.

Parte 1 – Cálculo da composição *ni* dos gases de combustão

Sistema de equações iniciais

Para a primeira parte, utilizou-se o seguinte sistema de equações, com **as** sendo o coeficiente estequiométrico, **phi** a razão de equivalência e **P** a pressão em atm:

$$a = (y_1 + y_5)N \quad (1)$$

$$b = (2y_2 + 2y_6 + y_7 + y_9)N \quad (2)$$

$$c + 2as/\phi = (2y_1 + y_2 + 2y_4 + y_5 + y_8 + y_9 + y_{10})N \quad (3)$$

$$d + 7.52as/\phi = (2y_3 + y_{10})N \quad (4)$$

$$K_1 = \frac{y_7 P^{1/2}}{y_6^{1/2}} \quad (5)$$

$$K_2 = \frac{y_8 P^{1/2}}{y_4^{1/2}} \quad (6)$$

$$K_3 = \frac{y_9}{y_4^{1/2} y_6^{1/2}} \quad (7)$$

$$K_4 = \frac{y_{10}}{y_4^{1/2} y_3^{1/2}} \quad (8)$$

$$K_5 = \frac{y_2}{y_4^{1/2} y_6 P^{1/2}} \quad (9)$$

$$K_6 = \frac{y_1}{y_4^{1/2} y_5 P^{1/2}} \quad (10)$$

$$\sum y_i = 1 \quad (11)$$

Sendo, portanto, um sistema de 11 equações com 11 incógnitas que são as frações molares **y1, y2, y3, y4, y5, y6, y7, y8, y9** e **y10** (referentes aos gases CO₂, H₂O, N₂, O₂, CO, H₂, H, O, OH, NO, respectivamente) e também o número total de mols da mistura **N**.

Onde as 4 primeiras equações são referentes ao balanço realizado a partir do número de átomos **a, b, c** e **d**, referentes ao Carbono, Hidrogênio, Oxigênio e Nitrogênio do combustível. E, da equação 5 a 10, fazem o uso da constante de equilíbrio **K** que é determinada a partir da equação (12) e tabela abaixo. Já (11) é uma somatória das frações molares **y** sendo, portanto, igual a 1.

$$\log_{10} K_i(T) = A_i * \ln\left(\frac{T}{1000}\right) + \frac{B_i}{T} + C_i + D_i T + E_i T^2 \quad (12)$$

K_i	A_i	B_i	C_i	D_i	E_i
K_1	+0.432168E + 00	-0.112464E + 05	+0.267269E + 01	-0.745744E - 04	+0.242484E - 08
K_2	+0.310805E + 00	-0.129540E + 05	+0.321779E + 01	-0.738336E - 04	+0.344645E - 08
K_3	-0.141784E + 00	-0.213308E + 04	+0.853461E + 00	+0.355015E - 04	-0.310227E - 08
K_4	+0.150879E - 01	-0.470959E + 04	+0.646096E + 00	+0.272805E - 05	-0.154444E - 08
K_5	-0.752364E + 00	+0.124210E + 05	-0.260286E + 01	+0.259556E - 03	-0.162687E - 07
K_6	-0.415302E - 02	+0.148627E + 05	-0.475746E + 01	+0.124699E - 03	-0.900227E - 08

Tabela 1 – Coeficientes para a constante de equilíbrio K

Equacionamento utilizado

Como não é possível isolar as variáveis (visto que estão se multiplicando entre si e têm termos com potência), trata-se de um sistema de equações não-lineares. Para tanto, com a ideia de agregar um chute inicial para as incógnitas mantendo uma proporção entre as mesmas, tem-se a necessidade de trabalhar algebricamente as equações afim de eliminar \mathbf{N} , visto que essa tem sua magnitude muito maior ao comparada aos valores de y_i que, por sua vez, deve estar entre 0 e 1. Com o coeficiente estequiométrico dado por:

$$as = a + \frac{b}{4} - \frac{c}{2} \quad (13)$$

Dessa forma, pode-se dividir as 4 equações iniciais (que contêm \mathbf{N}) por qualquer uma das equações (1), (2), (3) e (4). Assim, dividindo por a , obtém-se:

$$\frac{a}{a} = \frac{(y_1+y_5)N}{(y_1+y_5)N} \quad (14)$$

$$\frac{b}{a} = \frac{(2y_2+2y_6+y_7+y_9)N}{(y_1+y_5)N} \quad (15)$$

$$\frac{c+2as/\phi}{a} = \frac{(2y_1+y_2+2y_4+y_5+y_8+y_9+y_{10})N}{(y_1+y_5)N} \quad (16)$$

$$\frac{d+7.52as/\phi}{a} = \frac{(2y_3+y_{10})N}{(y_1+y_5)N} \quad (17)$$

Assim, elimina-se a incógnita \mathbf{N} e também a equação (1) para evitar redundância algébrica, mantendo um sistema de 10 incógnitas e 10 variáveis. E para eliminar a dependência do **log10**, pode-se utilizar sua inversa na equação (12) afim de obter unicamente \mathbf{K} a partir da equação:

$$Ki(T) = 10^{(Ai * \ln\left(\frac{T}{1000}\right) + \frac{Bi}{T} + Ci + DiT + ET^2)} \quad (18)$$

Além disso, para retirar algumas dependências de raízes nas equações (5) a (10), elevou-se ao quadrado afim de manipular novamente as equações para que o comando **fsolve** possa convergir.

Dessa maneira, pode-se utilizar o comando **fsolve** do Scilab para obter a solução do sistema. Logo, ao obter os valores de y_i , pode-se utilizar as seguintes equações para encontrar o valor **N** e **ni** respectivamente:

$$N = \frac{a}{y_1 + y_5} \quad (19)$$

$$ni = N * y_i \quad (20)$$

As condições lógicas e os chutes iniciais

O comando utilizado (**fsolve**) necessita de aprimoramento para se alcançar os valores ideais para o problema em questão. Isso porque os resultados para um mesmo problema podem mudar muito conforme variar os chutes iniciais para cada valor de **y**.

Por exemplo, para validar o código foi utilizado o arquivo em PDF da P4 disponível no EAD que já consta os valores de fração molar para o problema em questão (gasolina, $T = 3000 \text{ K}$, $P = 5000 \text{ kPa}$, $\phi = 0,8$, etc.). Portanto, pode-se chutar o valor inicial para o **fsolve**, os próprios resultado do documento P4. Obtendo resultados para **y** iguais aos do arquivo. Todavia, caso deseja-se alterar as condições de entrada (temperatura, pressão, combustível, ϕ) o código já não será válido, visto que os chutes iniciais podem não ser próximos o suficiente para uma outra condição. Ou seja, tal código só serviria para esse caso. Por conta disso, o código deve ser aprimorado para que possa ser generalizado.

O **fsolve** fornece 3 dados de saída, sendo:

- (I) o valor de **y** para a solução do problema;
- (II) o resultado da operação matemática ao substituir o valor de **y** encontrado nas equações especificadas em *function* (deve ser o mais próximo de zero possível);
- (III) e um critério de verificação para ver se o programa está convergindo ou não (indica 1, caso convirja).

Através de condições lógicas do problema de equilíbrio químico, tem-se as seguintes relações:

- Todo valor de **y** deve estar entre 0 e 1;
- A soma dos valores de **y** deve ser resultado no mais próxima de 1 possível;
- E que (III) (definida como variável “*in*” no código utilizado), deve ser igual a 1.

Dessa forma, pode-se fazer variar o chute inicial em um intervalo entre 0 e 1, afim de se obter o valor de chute inicial ideal na qual o **fsolve** converge respeitando as condições impostas do problema através de *if-elseif-end*.

Exemplo para validação do código

Ao desenvolver o código da **parte 1** (disponível em **APENDICES**) para encontrar as frações molares, pode-se utilizar o exemplo da P4, “*Temperatura Adiabática de Chama*”, onde utiliza-se a gasolina (C₇H₁₇) como combustível a temperatura de 3000 K e pressão de 5000 kPa, para validar o código afim de obter resultados próximos do documento.

Como mencionado anteriormente, os chutes iniciais devem ser próximos dos resultados esperados de fração molar. Portanto, utilizando o código da parte 1, pode-se variar o chute inicial encontrando os valores na qual a solução convirja. Com $\phi = 0.8$, encontra-se:

Gases de combustão	Fração molar y da P4	Fração molar y obtida com o código atual
CO ₂	0.0775	0.077568159
H ₂ O	0.1064	0.106415942
N ₂	0.7203	0.720165963
O ₂	0.0359	0.035867269
CO	0.0191	0.019073058
H ₂	0.0036	0.003629645
H	0.0013	0.001349214
O	0.0030	0.003030288
OH	0.0133	0.013259709
NO	0.0196	0.019640749

Tabela 2 – Comparativo entre frações molares para a gasolina

Como pode-se perceber, estimando um valor para ϕ , para alguns gases obtém-se valores bem próximos. Além disso, tem-se para a mistura uma composição molar, obtida com o código, dada abaixo:

Gases de combustão	Composição molar ni [mol]
CO ₂	5.618483
H ₂ O	7.708011
N ₂	52.16368
O ₂	2.597969
CO	1.381516
H ₂	0.262905
H	0.097727
O	0.219492
OH	0.960438
NO	1.422635
TOTAL	72.434

Tabela 3 – Composição molar ni obtida para a gasolina

Com a tabela, o número de mols total da mistura (**N**) é igual a 72.434 mols.

Parte 2 – Gráfico y_i x T para o metano

Para a segunda parte, adicionou-se um código para gerar 3 gráficos variando o **phi** em 0.8, 1 e 1.2 para o metano (CH_4), afim de avaliar o comportamento das frações molares dos gases de combustão para o as medidas de temperatura de 1000 a 3500 K, submetido a uma pressão de 50 bar.

Parte 2a - Cálculo dos chutes iniciais para um intervalo de temperatura

Com o intuito de obter o gráfico de y x T , se faz necessário obter (assim como na parte 1) valores de chutes iniciais na qual faz o código convergir em um dado intervalo de temperatura. Assim, foi utilizado o código da **parte 2a** (disponível em **APÊNDICES**) onde é possível relacionar um valor de chute inicial com um valor de temperatura do intervalo proposto através do *linspace*. Isto é:

- Para **T(1)** tem-se um chute inicial **i_2(1)**;
- Para **T(2)** tem-se um chute inicial **i_2(2)**;
- Para **T(3)** tem-se um chute inicial **i_2(3)**;
- E assim por diante.

Onde os **i_2** foi definida para ser o chute inicial onde o código converge para aquela temperatura que, para cada caso, deve ser adicionada ao código a seguir.

Parte 2b - Cálculo dos chutes iniciais para um intervalo de temperatura

Dessa forma, para cada situação de **phi**, foi possível obter um vetor de **i_2** que é relacionado com o intervalo de temperatura, sendo possível plotar os gráficos com o código **parte 2b** (disponível em **APÊNDICES**). Logo:

- Para $\phi = 0.8$, tem-se o seguinte vetor de convergência e gráfico:
 $i_2 = [0.00805 \ 0.0149 \ 0.00448 \ 0.00444 \ 0.00455 \ 0.00452 \ 0.00455 \ 0.00931 \ 0.00105 \ 0.00135 \ 0.00138 \ 0.00182 \ 0.00258 \ 1.9\text{E-}4 \ 3.0\text{E-}4 \ 0.0448 \ 0.00239 \ 2.0\text{E-}4 \ 4.2\text{E-}4 \ 0.00184 \ 2.0\text{E-}4 \ 8.0\text{E-}5 \ 0.1245 \ 2.0\text{E-}4 \ 2.3\text{E-}4 \ 4.7\text{E-}4 \ 3.1\text{E-}4 \ 2.0\text{E-}5 \ 2.6\text{E-}4 \ 9.0\text{E-}5 \ 4.7\text{E-}4 \ 1.1\text{E-}4 \ 0.0 \ 2.0\text{E-}5 \ 6.0\text{E-}5];$

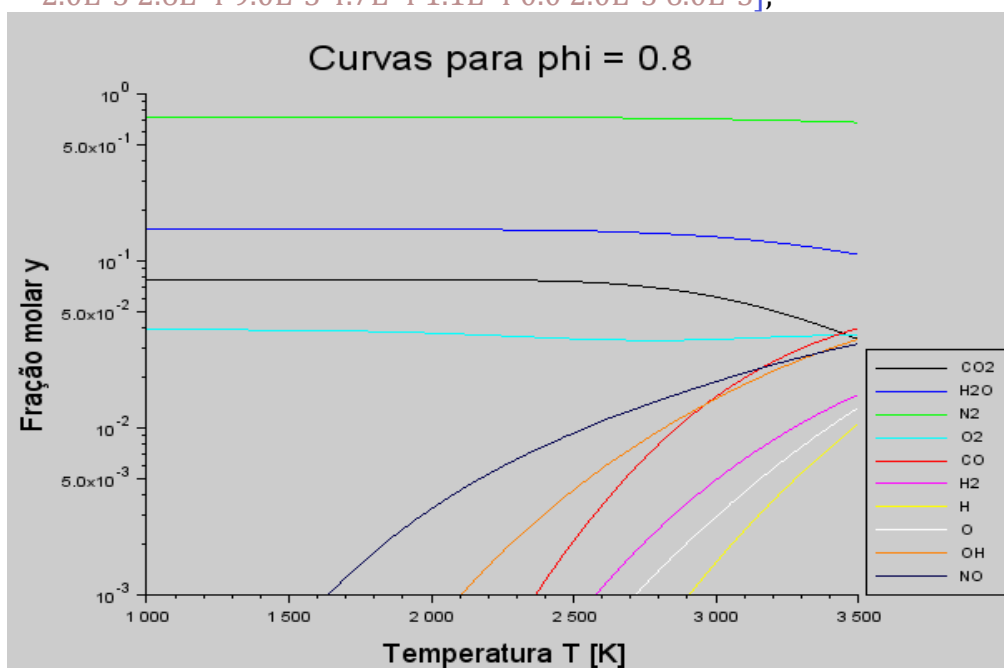


Gráfico 1 – Curvas para o metano com $\phi = 0.8$

- Para $\phi = 1$, tem-se o seguinte vetor de convergência e gráfico:

$i_2 = [1.8E-4 \ 6.9E-4 \ 3.8E-4 \ 3.0E-4 \ 2.9E-4 \ 7.1E-4 \ 9.0E-5 \ 2.5E-4 \ 2.0E-5 \ 4.6E-4 \ 6.0E-5 \ 3.5E-4 \ 1.6E-4 \ 2.3E-4 \ 1.0E-5 \ 2.0E-4 \ 2.0E-5 \ 3.0E-4 \ 1.3E-3 \ 9.0E-5 \ 5.8E-4 \ 0.0 \ 1.0E-5 \ 1.1E-4 \ 8.4E-4 \ 6.0E-4 \ 5.6E-4 \ 2.0E-5 \ 9.0E-5 \ 2.7E-4 \ 2.1E-4 \ 7.0E-5 \ 2.0E-4 \ 1.6E-4 \ 8.0E-5];$

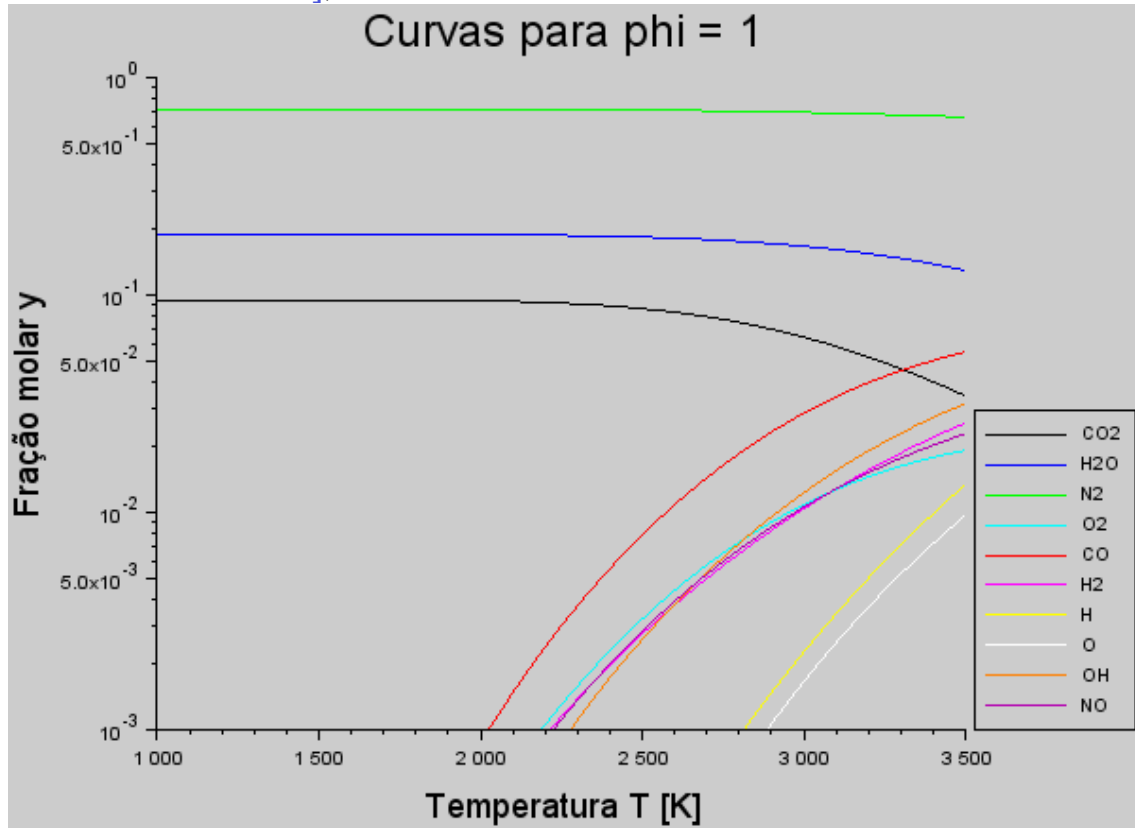


Gráfico 2 – Curvas para o metano com $\phi = 1$

- Para $\phi = 1.2$, tem-se o seguinte vetor de convergência e gráfico:

$i_2 = [0.00255 \ 0.5319 \ 0.01044 \ 0.01093 \ 0.01083 \ 0.01409 \ 0.01344$
 $0.00302 \ 0.00174 \ 0.00125 \ 0.00106 \ 8.3E-4 \ 0.00114 \ 8.5E-4 \ 9.9E-4 \ 0.0018$
 $7.4E-4 \ 6.0E-5 \ 2.0E-5 \ 3.0E-5 \ 3.1E-4 \ 3.0E-5 \ 1.2E-4 \ 2.0E-5 \ 5.0E-5 \ 2.0E-5$
 $1.0E-4 \ 4.0E-5 \ 3.7E-4 \ 2.9E-4 \ 2.7E-4 \ 2.3E-4 \ 2.3E-4 \ 4.0E-5 \ 4.2E-4];$

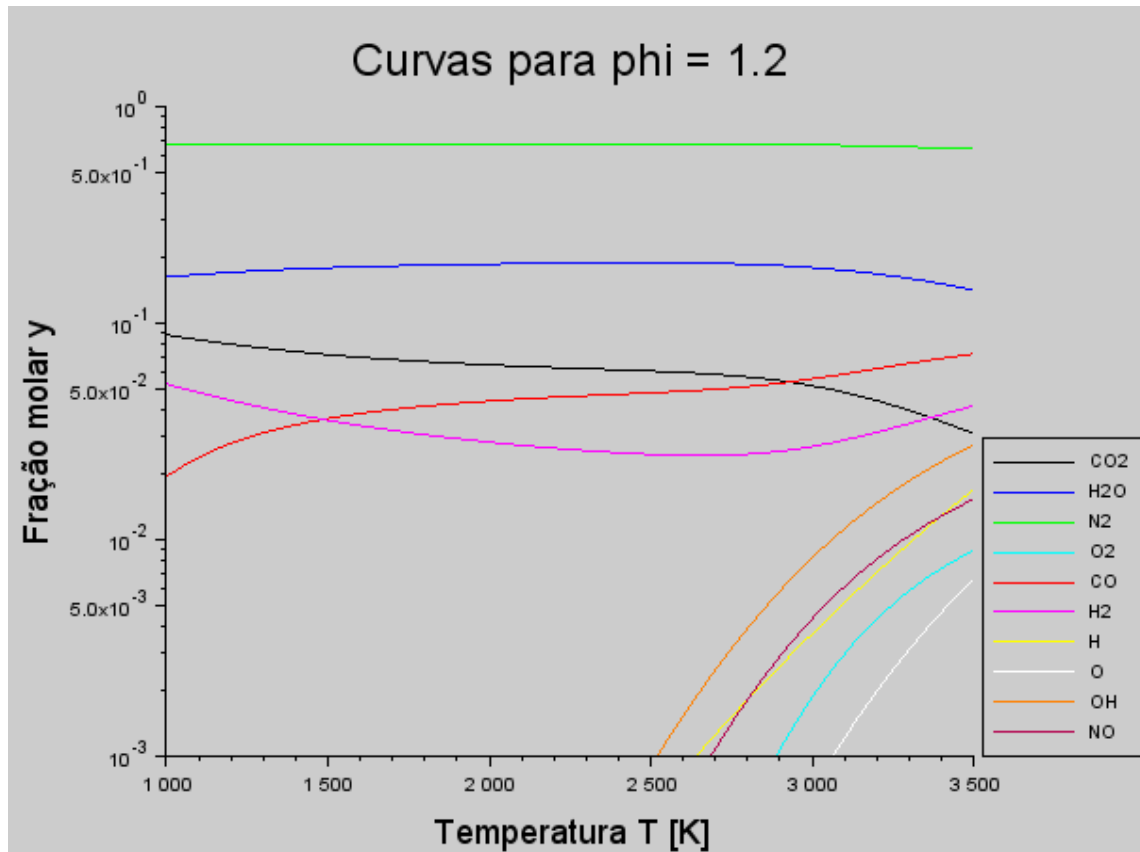


Gráfico 3 – Curvas para o metano com $\phi = 1.2$

APENDICES

Código da Parte 1 – Cálculo de ni

clc

clear

//Criado em 2 de novembro de 2018

//Engenheiro responsável: Adrian Beppu Hirata

//Engenheiro verificador: Carlos Renan Cândido da Silva

//PARTE 1 - P5 - Equilíbrio Químico

//Programa para calcular as composições ni de cada um dos seguintes produtos de combustão:

//CO₂, H₂O, N₂, O₂, CO, H₂, H, O, OH e NO, respectivamente, a uma dada temperatura, pressão e

//razão de equivalência phi.

//ARGUMENTOS DE ENTRADA

//Para o exemplo da gasolina com C₇H₁₇ (dados retirados da P1):

a = 7; //nº de átomos de carbono

b = 17; //nº de átomos de hidrogênio

c = 0; //nº de átomos de oxigênio

d = 0; //nº de átomos de nitrogênio

as = a + b/4 - c/2; //coeficiente estequiométrico

//Condições estabelecidas para a gasolina (dados retirados da P4):

phi = 0.8; //Razão de equivalência ACs/AC

T = 3000; //Temperatura em K

*P = 50*0.986923; //Pressão em atm*

//Matriz com as constantes para K

// A B C D E

*M = [0.432168*10⁰⁰ -0.112464*10⁰⁵ 0.267269*10⁰¹ -0.745744*10⁻⁰⁴ 0.242484*10⁻⁰⁸*

*0.310805*10⁰⁰ -0.129540*10⁰⁵ 0.321779*10⁰¹ -0.738336*10⁻⁰⁴ 0.344645*10⁻⁰⁸*

*-0.141784*10⁰⁰ -0.213308*10⁰⁴ 0.853461*10⁰⁰ 0.355015*10⁻⁰⁴ -0.310227*10⁻⁰⁸*

*0.150879*10⁻⁰¹ -0.470959*10⁰⁴ 0.646096*10⁰⁰ 0.272805*10⁻⁰⁵ -0.154444*10⁻⁰⁸*

*-0.752364*10⁰⁰ 0.124210*10⁰⁵ -0.260286*10⁰¹ 0.259556*10⁻⁰³ -0.162687*10⁻⁰⁷*

*-0.415302*10⁻⁰² 0.148627*10⁰⁵ -0.475746*10⁰¹ 0.124699*10⁻⁰³ -0.900227*10⁻⁰⁸];*

//Equação para o cálculo da constante Ki de equilíbrio químico

```
K = 10^(M(:,1).*log(T/1000)+M(:,2)./T+M(:,3)+M(:,4).*T+M(:,5).*(T^2));  
K_2 = [K.^2];
```

//SEÇÃO PARA ENCONTRAR VALOR DE CHUTES INICIAIS (i) QUE FAZEM O PROBLEMA CONVERGIR

```
for i=0:0.001:1;    //intervalo para os chutes iniciais
```

//Função para o sistema de equações não-lineares

```
function x=g(y);
```

```
    x(1) = b/a*(y(1)+y(5))-(2*y(2)+2*y(6)+y(7)+y(9));  
    x(2) = (c+2*as/phi)/a*(y(1)+y(5))-  
(2*y(1)+y(2)+2*y(4)+y(5)+y(8)+y(9)+y(10));  
    x(3) = (d+7.52*as/phi)/a*(y(1)+y(5))-(2*y(3)+y(10));  
    x(4) = y(1)+y(2)+y(3)+y(4)+y(5)+y(6)+y(7)+y(8)+y(9)+y(10)-1;  
    x(5) = K_2(1)*y(6)-y(7)*y(7)*P;  
    x(6) = K_2(2)*y(4)-y(8)*y(8)*P;  
    x(7) = K_2(3)*y(4)*y(6)-y(9)*y(9);  
    x(8) = K_2(4)*y(4)*y(3)-y(10)*y(10);  
    x(9) = K_2(5)*y(4)*y(6)*y(6)*P-y(2)*y(2);  
    x(10) = K_2(6)*y(4)*y(5)*y(5)*P-y(1)*y(1);
```

```
endfunction
```

//COMANDO PARA A SOLUÇÃO DO SISTEMA DE EQUAÇÕES

//y: valores de fração molar; fp: resultados ao substituir y nas equações;

// in: é o indicador de convergência, converge quando in = 1;

```
[y fp in] = fsolve([i,i,i,i,i,i,i,i],g,10^-15)
```

```
y=y'
```

//CONDIÇÕES LÓGICAS DE EQUILÍBRIO QUÍMICO PARA A SELEÇÃO DE FRAÇÃO MOLAR

//Condições caso y seja negativo (o que não pode)

```
if y(1)<0  
    y(:,1)=[]  
elseif y(2)<0  
    y(:,1)=[]  
elseif y(3)<0  
    y(:,1)=[]  
elseif y(4)<0  
    y(:,1)=[]
```

```

elseif y(5)<0
y(:,1)=[]
elseif y(6)<0
y(:,1)=[]
elseif y(7)<0
y(:,1)=[]
elseif y(8)<0
y(:,1)=[]
elseif y(9)<0
y(:,1)=[]
elseif y(10)<0
y(:,1)=[]
end

```

```

//Condições na convergência de fsolve e soma das frações = 1
if in == 1 & sum(y)==1;

```

```

//Cálculo do n° total de mols da mistura N
N = a/(y(1)+y(5)); //[mols]

```

```

//Cálculo da composição de cada um dos produtos de combustão
//CO2, H2O, N2, O2, CO, H2, H, O, OH e NO, respectivamente
ni = N.*y;      //[mols]

```

```

disp(['Fração molar' 'Composição molar [mol]'])
disp([ y      ni])

```

```

break

```

```

end

```

```

end

```

Código da Parte 2a – Cálculo do vetor de convergência i_2

```
clc  
clear
```

```
//Criado em 12 de novembro de 2018  
//Engenheiro responsável: Adrian Beppu Hirata  
//Engenheiro verificador: Carlos Renan Cândido da Silva
```

```
//PARTE 2a - P5 - Equilíbrio Químico  
//Programa para encontrar o vetor de convergência i_2 para um dado  
intervalo de temperatura
```

```
//ARGUMENTOS DE ENTRADA  
//Para o metano com CH4 (dados retirados da P1):  
a = 1;           //nº de átomos de carbono  
b = 4;           //nº de átomos de hidrogênio  
c = 0;           //nº de átomos de oxigênio  
d = 0;           //nº de átomos de nitrogênio  
as = a + b/4 - c/2; //coeficiente estequiométrico
```

```
//Condições estabelecidas para o metano:  
phi = 0.8;       //Razão de equivalência ACs/AC  
T = linspace(1000,3500,35); //Temperatura em K  
P = 50*0.986923; //Pressão em atm
```

```
//Matriz com as constantes para K  
//      A          B          C          D          E  
M = [0.432168*10^00 -0.112464*10^05 0.267269*10^01 -0.745744*10^-  
04 0.242484*10^-08  
0.310805*10^00 -0.129540*10^05 0.321779*10^01 -0.738336*10^-04  
0.344645*10^-08  
-0.141784*10^00 -0.213308*10^04 0.853461*10^00 0.355015*10^-04  
-0.310227*10^-08  
0.150879*10^-01 -0.470959*10^04 0.646096*10^00 0.272805*10^-05  
-0.154444*10^-08  
-0.752364*10^00 0.124210*10^05 -0.260286*10^01 0.259556*10^-03  
-0.162687*10^-07  
-0.415302*10^-02 0.148627*10^05 -0.475746*10^01 0.124699*10^-03  
-0.900227*10^-08];
```

```
for k = 1:length(T);
```

```
//Equação para o cálculo da constante Ki de equilíbrio químico
```

```

K =
10^(M(:,1).*log(T(k)/1000)+M(:,2)./T(k)+M(:,3)+M(:,4).*T(k)+M(:,5).*(T(k)
^2));
K_2 = [K.^2];

```

//Função para o sistema de equações não-lineares

for i_2=0:0.00001:1; //intervalo possível para i

function x=g(y)

```

x(1) = b/a*(y(1)+y(5))-(2*y(2)+2*y(6)+y(7)+y(9));
x(2) = (c+2*as/phi)/a*(y(1)+y(5))-
(2*y(1)+y(2)+2*y(4)+y(5)+y(8)+y(9)+y(10));
x(3) = (d+7.52*as/phi)/a*(y(1)+y(5))-(2*y(3)+y(10));
x(4) = y(1)+y(2)+y(3)+y(4)+y(5)+y(6)+y(7)+y(8)+y(9)+y(10)-1;
x(5) = K_2(1)*y(6)-y(7)*y(7)*P;
x(6) = K_2(2)*y(4)-y(8)*y(8)*P;
x(7) = K_2(3)*y(4)*y(6)-y(9)*y(9);
x(8) = K_2(4)*y(4)*y(3)-y(10)*y(10);
x(9) = K_2(5)*y(4)*y(6)*y(6)*P-y(2)*y(2);
x(10) = K_2(6)*y(4)*y(5)*y(5)*P-y(1)*y(1);

```

endfunction

//Comando para a solução do sistema de equações

```
[y fv in] = fsolve([i_2,i_2,i_2,i_2,i_2,i_2,i_2,i_2,i_2,i_2],g,10^-15)
```

y=y'

//CONDIÇÕES LÓGICAS DE EQUILÍBRIO QUÍMICO PARA A SELEÇÃO DE FRAÇÃO MOLAR

//Condições caso y seja negativo (o que não pode)

```

if y(1)<0
y(:,1)=[]
elseif y(2)<0
y(:,1)=[]
elseif y(3)<0
y(:,1)=[]
elseif y(4)<0
y(:,1)=[]
elseif y(5)<0
y(:,1)=[]
elseif y(6)<0

```

```

y(:,1)=[]
elseif y(7)<0
y(:,1)=[]
elseif y(8)<0
y(:,1)=[]
elseif y(9)<0
y(:,1)=[]
elseif y(10)<0
y(:,1)=[]
end

```

```

//Condições na convergência de fsolve e soma das frações próxima de 1
if in == 1 & 0.99<=sum(y) & sum(y)<=1.1;

```

```

T_i_2 = [T(k) i_2] //Relação entre cada temperatura com o valor de
convergência

```

```

disp(T_i_2) //Mostra o vetor a ser copiado no Console do Scilab

```

```

break

```

```

end

```

```

end

```

```

end

```

Código da Parte 2b – Gráfico y x T utilizando o vetor de convergência i_2

```
clc
clear
```

```
//Criado em 12 de novembro de 2018
```

```
//Engenheiro responsável: Adrian Beppu Hirata
```

```
//Engenheiro verificador: Carlos Renan Cândido da Silva
```

```
//PARTE 2b - P5 - Equilíbrio Químico
```

```
//Programa para gerar o gráfico de fração molar y x Temperatura T [K] para as curvas de:
```

```
//CO2, H2O, N2, O2, CO, H2, H, O, OH e NO, em um intervalo de temperatura (1000 a 3500 K),
```

```
//a pressão de 50 bar e a uma dada razão de equivalência phi para o metano, utilizando
```

```
//o vetor de convergência i_2 encontrado em 2a
```

```
//ARGUMENTOS DE ENTRADA
```

```
//Para o metano com CH4 (dados retirados da P1):
```

```
a = 1; //nº de átomos de carbono
```

```
b = 4; //nº de átomos de hidrogênio
```

```
c = 0; //nº de átomos de oxigênio
```

```
d = 0; //nº de átomos de nitrogênio
```

```
//Vetor de convergência i_2 encontrado na parte 2a
```

```
i_2 = [0.00805 0.0149 0.00448 0.00444 0.00455 0.00452 0.00455 0.00931  
0.00105 0.00135 0.00138 0.00182 0.00258 1.9E-4 3.0E-4 0.0448 0.00239  
2.0E-4 4.2E-4 0.00184 2.0E-4 8.0E-5 0.1245 2.0E-4 2.3E-4 4.7E-4 3.1E-4  
2.0E-5 2.6E-4 9.0E-5 4.7E-4 1.1E-4 0.0 2.0E-5 6.0E-5];
```

```
as = a+b/4-c/2; //Coeficiente estequiométrico
```

```
phi = 0.8; //Razão de equivalência ACs/AC
```

```
T = linspace(1000,3500,35); //Intervalo de temperatura em K
```

```
P = 50*0.986923; //Pressão em atm
```

```
//Matriz M com as constantes para Ki
```

```
// A B C D E
```

```
M = [0.432168*10^00 -0.112464*10^05 0.267269*10^01 -0.745744*10^-04 0.242484*10^-08  
0.310805*10^00 -0.129540*10^05 0.321779*10^01 -0.738336*10^-04 0.344645*10^-08  
-0.141784*10^00 -0.213308*10^04 0.853461*10^00 0.355015*10^-04 -0.310227*10^-08  
0.150879*10^-01 -0.470959*10^04 0.646096*10^00 0.272805*10^-05 -0.154444*10^-08
```



```

-0.752364*10^00 0.124210*10^05 -0.260286*10^01 0.259556*10^-03
-0.162687*10^-07
-0.415302*10^-02 0.148627*10^05 -0.475746*10^01 0.124699*10^-03
-0.900227*10^-08];

```

//Função para o sistema de equações não-lineares

```
function [x]=g(y)
```

```

x(1) = b/a*(y(1)+y(5))-(2*y(2)+2*y(6)+y(7)+y(9));
x(2) = (c+2*as/phi)/a*(y(1)+y(5))-
(2*y(1)+y(2)+2*y(4)+y(5)+y(8)+y(9)+y(10));
x(3) = (d+7.52*as/phi)/a*(y(1)+y(5))-(2*y(3)+y(10));
x(4) = y(1)+y(2)+y(3)+y(4)+y(5)+y(6)+y(7)+y(8)+y(9)+y(10)-1;
x(5) = K_2(1)*y(6)-y(7)*y(7)*P;
x(6) = K_2(2)*y(4)-y(8)*y(8)*P;
x(7) = K_2(3)*y(4)*y(6)-y(9)*y(9);
x(8) = K_2(4)*y(4)*y(3)-y(10)*y(10);
x(9) = K_2(5)*y(4)*y(6)*y(6)*P-y(2)*y(2);
x(10) = K_2(6)*y(4)*y(5)*y(5)*P-y(1)*y(1);

```

```
endfunction
```

```
for p=1:length(T);
```

//Equação para o cálculo da constante Ki de equilíbrio químico

```

K =
10^(M(:,1).*log(T(p)/1000)+M(:,2)./T(p)+M(:,3)+M(:,4).*T(p)+M(:,5).*T(p).
^2);
K_2 = [K.^2];

```

//Comando para a solução do sistema de equações

```

[y(p,:)] =
[fsolve([i_2(p),i_2(p),i_2(p),i_2(p),i_2(p),i_2(p),i_2(p),i_2(p),i_2(p),g]
;
end

```

//Gráfico de fração molar y por Temperatura T [K] em escala logarítmica no eixo de y

```
figure()
```

```
T=T' //Intervalo da temperatura no eixo de x
```

```
plot2d1("onl",T,y) //Plot do gráfico
```

```
zoom_rect([1000,10^-3,3500,10^0]) //Zoom no gráfico
```

//Legendas para as curvas e eixos

```
h1 = legend(['CO2' 'H2O' 'N2' 'O2' 'CO' 'H2' 'H' 'O' 'OH' 'NO'],-4);  
xlabel("Temperatura T [K]","fontsize",4)  
ylabel("Fração molar y","fontsize",4)  
  
//Título do gráfico  
titulo = ["Curvas para phi = 0.8"];  
title(titulo,"fontsize",5)
```